

Abstract of Paper Proposed for Presentation at the
41st AIAA Applied Aerodynamics Conference
Orlando, FL
June, 2003

Cartesian-Grid Simulations of a Canard-Controlled Missile with a Spinning Tail

Scott M. Murman
ELORET
MS T27B
Moffett Field, CA 94035
smurman@nas.nasa.gov

Michael J. Aftosmis
NASA Ames Research Center
MS T27B
Moffett Field, CA 94035
aftosmis@nas.nasa.gov

Abstract

Technical Topic: Missile aerodynamics, moving-body simulations, Cartesian mesh

The proposed paper presents a series of simulations of a geometrically complex, canard-controlled, supersonic missile with free-spinning tail fins. Time-dependent simulations were performed using an inviscid Cartesian-grid-based method with results compared to both experimental data and high-resolution Navier-Stokes computations. At fixed free stream conditions and canard deflections, the tail spin rate was iteratively determined such that the net rolling moment on the empennage is zero. This rate corresponds to the time-asymptotic rate of the free-to-spin fin system. After obtaining spin-averaged aerodynamic coefficients for the missile, the investigation seeks a fixed-tail approximation to the spin-averaged aerodynamic coefficients, and examines the validity of this approximation over a variety of freestream conditions.

1 Introduction

Over the past decade, static Computational Fluid Dynamics (CFD) simulations over increasingly complex vehicles have become commonplace. In this evolution, non-body-fitted Cartesian grid methods have proven to be particularly useful for

automatically meshing geometrically complex vehicles[1-6]. Recently this class of meshing and solution techniques has been extended to dynamic simulations[7-10], where components of the geometry move in some manner *during* the simulation. This makes highly-automated simulations of complex three-dimensional vehicles with components in relative motion more feasible. The proposed paper adopts a non-body-fitted Cartesian method to study the performance of a supersonic, canard-controlled missile with a free-spinning tail. On this type of vehicle, the tail fins are free to spin as a unit around the missile longitudinal axis. As a result, torque from aerodynamic loads on the empennage cause the fin system to spin, even under steady-state flight conditions.

Missiles with dynamic components can pose significant challenges for numerical simulation. The spinning tail is a by-product of the forces on the missile and is integral to its aerodynamic performance. Nevertheless, the performance of the missile is primarily characterized by the spin-averaged aerodynamic coefficients, and hence time-dependent, moving-body simulations are required to predict even *static* stability and control (S&C) information. Moreover, the spin-rate of the fin system is governed not only by the wind vector and canard deflections, but also by the strength and location of the convected canard vortices, whose induced velocity field differentially loads the tail fins at low angles of attack. The need to convect the canard vortices over the length of the missile to interact with the fin system directly impacts the size of the computational mesh required for accurate numerical simulation. The combination of these factors makes for CPU-intensive simulations since the physics requires both highly-resolved spatial grids and time-dependent, moving-body solution methodologies.

The proposed paper details the missile geometry under consideration, important features of the computational mesh, and the numerical method used for the simulations. The numerical investigations first simulate the missile with the fins fixed at various azimuths around the missile axis to establish a zero-spin-rate baseline. Dynamic simulations are then performed with an imposed spin rate on the tail. An iterative process is used to determine the spin rate which predicts a zero spin-averaged torque on the tail. The final simulations will be compared with both experimental data and highly-resolved Navier-Stokes numerical simulations. The final paper will also extend the analysis by comparison of the dynamic and fixed-tail (static) results in an attempt to find a static fixed-tail approximation to the spin-averaged coefficients of the dynamic simulations. If such a correlation exists, it would dramatically decrease the expense of future studies of missiles with free-spinning tails.

2 Numerical Method

2.1 Geometry and Computational Mesh

Figure 1 shows three views of the canard-controlled missile used for this study. The missile is depicted with the tail in the + position, corresponding to a tail rotation angle, $\phi_{tail} = 45^\circ$. Zero rotation angle is obtained when the tail is in the \times position, and positive rotation is clockwise when observed from the missile nose. The canards are shown in their deflected position – pitched asymmetrically with $\delta_c = 16^\circ$ to command a starboard yaw of the missile. The missile body has a cylindrical cross-section with a fineness ratio (length/diameter ratio) of about 15.0. Two conduits, which are raised off the body and anchored at regular intervals leaving a small gap, run the length of the body. In addition, the missile has a ring of 10 bluff protuberances at roughly the mid-station of the body, and several others at the aft end near the free-spinning tail fins. The leading edges of the fins extend forward along the body, and the root station is cutaway to clear both the conduits and other hardware as the fins sweep over the missile body. The minimum clearance is about 1/8th of the fin thickness, and occurs when the fin passes over the protuberances on the aft missile body.

Detailed enlargements of the surface triangulation near the nose, mid-body and tail fins are shown in Fig. 2. The tail is in the + position which puts the upper and lower fins over two of the conduits and the other two fins over protuberances in the aft missile forebody. This triangulation was produced directly from CAD solids using the software described in [11]. This software uses the CAPRI library[12, 13] to access the CAD geometry using the CAD systems' native query routines and geometry engine. The final triangulation (shown) uses approximately 400,000 triangles which were used as input to the mesh generation system[5].

Figure 3 shows the non-body-fitted baseline Cartesian mesh used for the simulations. This figure shows the tail fin in the \times position ($\phi_{tail} = 0^\circ$), and the mesh is displayed by several cutting planes behind and perpendicular to the missile axis. In computing flows around canard-controlled missiles, it can be very important to avoid excessively dissipating the canard vortices as they convect the length of the missile body. To provide this resolution, the mesh has a pre-specified adaptation region covering the entire missile, and within this region the mesh is refined 3 levels further driven by surface curvature as described in [5]. In addition to the canard vortices, a pre-specified adaptation region is designed to capture the shocks generated by many of the surface features on the missile body. Since streamlines passing through these shocks will impact the spinning fins, resolution and propagation of these shocks may be important. Resolution requirements for the baseline mesh were established using guidelines from previous simulations of canard-controlled missiles[9], and by performing a mesh resolution study with the current geometry. As the missile tail spins over the course of the simulation the mesh responds to track the body motion, re-adapting to the new geometry at each timestep (cf. [9, 10]). The snapshot shown here has ap-

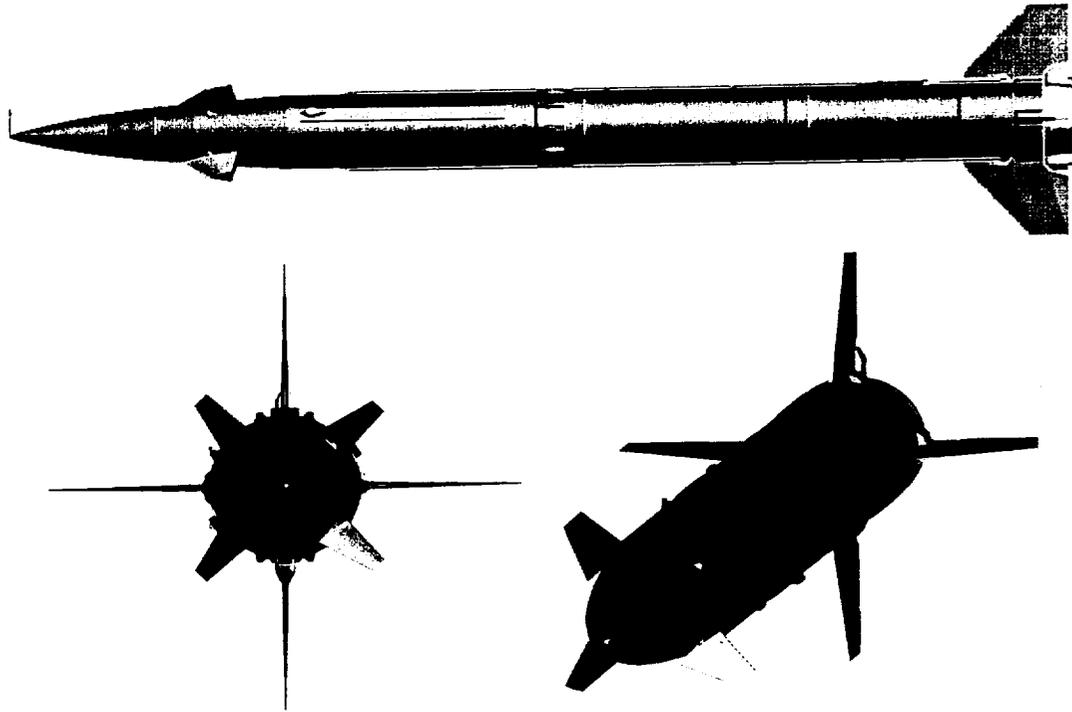


Figure 1: Front, side and isometric view of generic missile with free-spinning tail. For the simulations presented, all 4 canards are deflected $\delta_c = 16^\circ$ to command a starboard yaw of the missile. The tail is shown in the + configuration which corresponds to a tail rotation angle, $\phi_{tail} = 45^\circ$. $\phi_{tail} = 0^\circ$ is obtained with the empennage in the \times position.

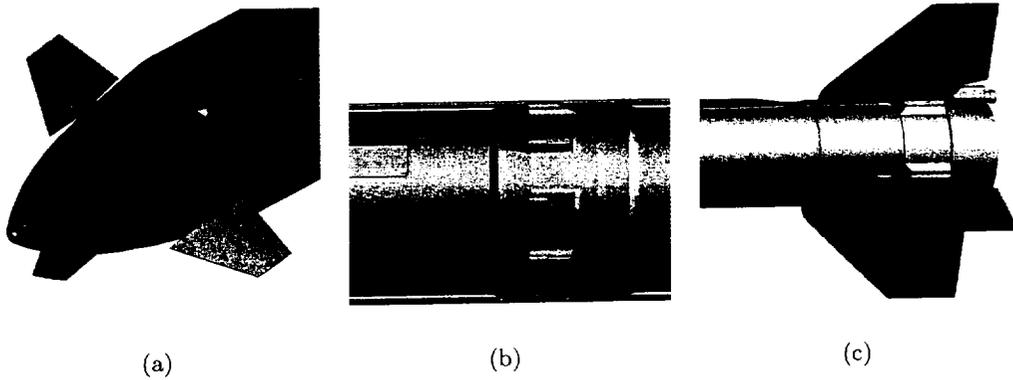


Figure 2: Surface mesh detail of missile configuration, 400,000 triangles.

proximately 4 million cells, and this total number of cells remains roughly constant over the course of a dynamic simulation with the tail section spinning.

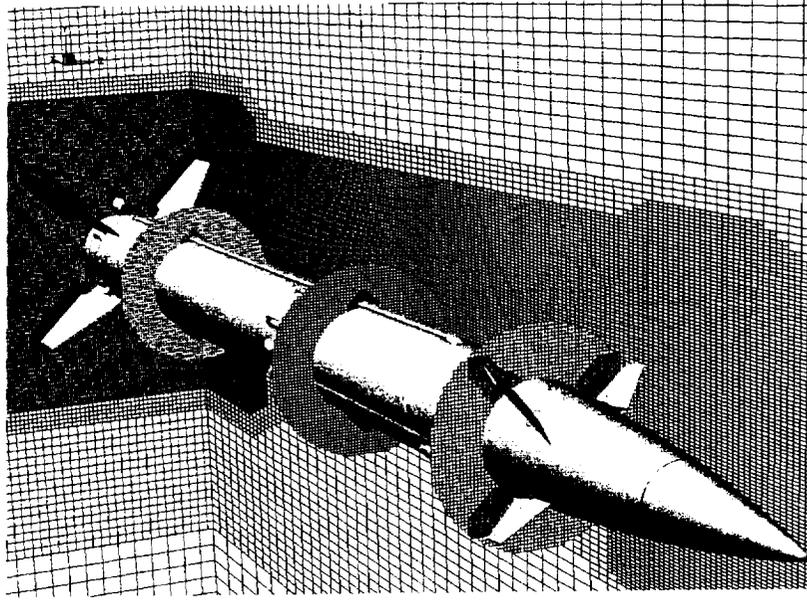


Figure 3: Cutting planes through the non-body-fitted Cartesian mesh used in simulations. Missile is shown with fins in the \times position and canards deflected asymmetrically $\delta_c = 16^\circ$. 4M Cartesian cells.

2.2 Cartesian Moving-Body Flow Solver

In order to simulate a missile with a spinning tail section, a scheme that allows rigid bodies to move relative to each other during a simulation is needed. A general numerical scheme for solving time-dependent flows with (optional) rigid-body motion for unstructured Cartesian meshes was developed from the parallel, steady-state solver described in [14].

2.2.1 Dual-time formulation

Extension of the steady-state flow solver to time-dependent flows was accomplished using a dual-time formulation (cf. Refs. [15, 16]),

$$\begin{aligned} \frac{d\mathbf{Q}}{d\tau} + R^*(\mathbf{Q}) &= 0 \\ R^*(\mathbf{Q}) &= \frac{\partial \mathbf{Q}}{\partial t} + R(\mathbf{Q}) \end{aligned} \tag{1}$$

where τ is referred to here as “pseudo-time”, and is the iterative parameter, and t is the physical time. \mathbf{Q} is the vector of conserved variables, and $R(\mathbf{Q})$ is an appropriate numerical quadrature of the flux divergence, $\frac{1}{V} \oint_S \mathbf{f} \cdot \mathbf{n} dS$. As $\frac{d\mathbf{Q}}{d\tau} \rightarrow 0$ the time-dependent formulation is recovered. The parallel multi-grid solver described in [14] is used to efficiently converge the inner pseudo-time integration. This is similar to

the scheme outlined by Jameson[17], however, the semi-implicit approach of Melson et al.[18] is used here for the physical time-derivative term.

Various time-dependent schemes can be constructed for Eqn. 1 by appropriately discretizing the time derivative. In the current work, it's desirable to use an unconditionally-stable, implicit scheme to allow a large timestep to be chosen based upon physical considerations rather than a potentially smaller stability-limited timestep. In the Cartesian embedded-boundary scheme, the cut-cell polyhedra can have arbitrarily small volumes, and a stability limit can be very restrictive. Using a large timestep also reduces the amount of computational work required to process the moving geometry and mesh through a complete simulation. In the current work, the backward Euler and 2nd-order backward time-integration schemes have both been utilized. Full details of the temporal discretization scheme are presented in Ref. [10].

2.2.2 Relative motion

Figure 4 shows a schematic of a rigid-body moving through a fixed Cartesian mesh over one discrete timestep. Cells cut at the beginning and end of the timestep are outlined in black, and the shaded region highlights cells which have been "swept" by the body through the timestep. These swept cells change volume and shape over the timestep, and can appear or disappear (or both) as well. Away from the swept region, the cells don't change and therefore require no special treatment. The swept-cells, however constitute the major challenge since the deformation of these cells over the timestep needs to be taken into account in order to satisfy the governing equations. The equations of motion for the deforming cells can be written in an integral conservation form as

$$\int_{V(t)} \mathbf{Q} dV = \left[- \oint_{S(t)} \mathbf{f} \cdot \mathbf{n} dS \right] dt \quad (2)$$

Integrating Eqn. 2 using the backward Euler scheme gives

$$\frac{\mathbf{Q}^{n+1} - \frac{V^n}{V^{n+1}} \mathbf{Q}^n}{\Delta t} = - \frac{1}{V^{n+1}} \left[\sum \bar{\mathbf{f}} \cdot \mathbf{n} \Delta S \right]^{n+1} \quad (3)$$

This can be numerically integrated using the dual-time scheme outlined above, and the term $\frac{V^n}{V^{n+1}} \mathbf{Q}^n$ becomes a fixed source term in the dual-time scheme. However, \mathbf{Q}^n is only available on the mesh at time level n , while it is required on the mesh at time level $n + 1$ in order to integrate Eqn. 3. Rewriting Eqn. 3 gives

$$\frac{\mathbf{Q}^{n+1} - \hat{\mathbf{Q}}^n}{\Delta t} = - \frac{1}{V^{n+1}} \left[\sum \bar{\mathbf{f}} \cdot \mathbf{n} \Delta S \right]^{n+1} \quad (4)$$

where $\hat{\mathbf{Q}}^n$ represents the state vector at time level n on the mesh at time level $n + 1$.

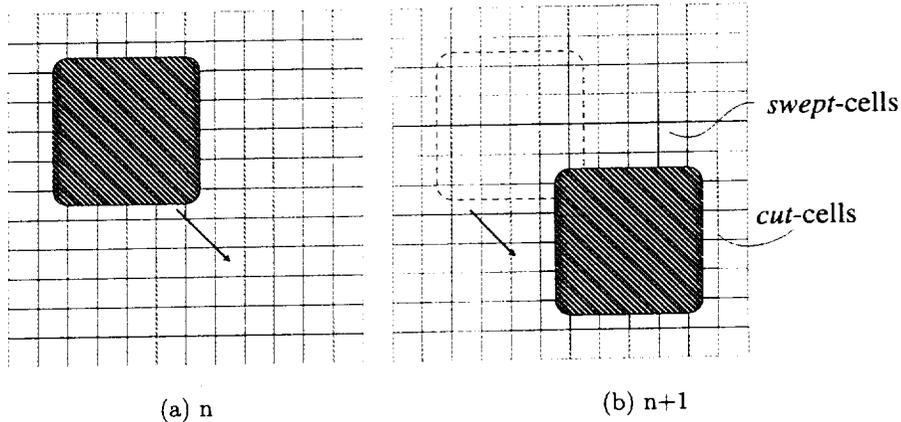


Figure 4: Schematic of a ridged body moving through a Cartesian mesh. Cells cut by the geometry at each timestep are shown in black, and cells swept by the geometry over the timestep are tinted yellow.

In the current scheme, the vector \mathbf{Q}^n is “mapped” from the mesh at time level n to the new mesh at $n + 1$ using an interpolation operator I_n^{n+1} .

$$\hat{\mathbf{Q}}^n = I_n^{n+1} \mathbf{Q}^n \quad (5)$$

If $I_n^{n+1} = \frac{V^n}{V^{n+1}}$ then Eqn. 3 is satisfied. The interpolation operator I_n^{n+1} can be determined exactly using a space-time approach (cf. Refs. [19, 20]), however doing so in three dimensions poses a problem in 4-D mesh generation. Instead, an approximate scheme is desired which maintains conservation away from the region of the relative motion. The current scheme determines I_n^{n+1} exactly for all cells away from the moving boundary, as well as the majority of the cut cells at both time levels. For a small minority of the swept cells I_n^{n+1} is approximated. In the current work, the mapping of the solution between two meshes is processed external to the flow solver with a single-pass algorithm. Note that since the motion is prescribed, all of the meshes can be processed a priori, and in parallel. Further details on the relative motion scheme can be found in [10].

3 Preliminary Numerical Results

The general 3-D Cartesian scheme outlined above is utilized to simulate the canard-controlled missile with spinning tail section described in Sec. 2.1. Since the flow conditions considered in this work are supersonic ($M_\infty = 1.6$, $\alpha = 4.0^\circ$), and the geometry upstream of the tail section is static, the flowfield within the tail section is periodic every 90° of spin. This periodicity was confirmed by the initial dynamic simulations. As such, it’s only necessary to simulate the motion of the tail section through 90° of rotation (after a small initial transient).

3.1 Static Baselines

In order to provide a baseline for comparison with dynamic, spinning-tail computations, a series of static, steady-state simulations with the tail fixed at various (non-uniform) azimuthal orientations were undertaken. Velocity magnitude contours at cutting planes along the longitudinal axis of the missile are shown with the fins in the \times position in Fig. 5, to highlight the convection of the canard vortices. The vortices shed from the tips of the NW, NE, and SE canards (following the compass directions viewed from the nose) are seen to convect down the length of the body. The vortex from the NE canard is stronger than the NW or SE vortices, as that canard is pitched up, while the other canards are pitched down. The vortex shed from the SW canard is “trapped” by the body as it convects upwards and dissipates. The asymmetric pitch of the canards causes an induced velocity which merges the vortices into the NE quadrant, where they impact the tail section. With the tail in the \times position, two vortices provide suction to the leeward face of the fin located in the NE quadrant, while the vortex from the SE canard provides suction to the windward face. This can be seen quantitatively in Fig. 6, where the variation of tail rolling moment, $(C_l)_{tail}$, with angle of rotation from the \times position is shown. A positive tail rolling moment would cause the tail section to rotate clockwise when viewed from the nose. As the tail fin encounters the (strongest) NE canard vortex, the tail rolling moment is at a maximum (near $\phi_{tail} = 80^\circ$). As the tail section moves to the $+$ position ($\phi_{tail} = 45^\circ$), the fins are farthest from the strong NE canard vortex, and evenly split (vortex suction inducing both CW and C-CW rotation) between the remaining two vortices, and the tail rolling moment is at a minimum. This differential pressure on the tail fin in the NE quadrant due to the vortices, along with the effects of dynamic pressure and angle of attack, and the induced velocity field from the canards all combine to cause the tail section to spin.

3.2 Dynamic Computations with Spinning Tail Section

The rotation rate of the tail section at the current flow conditions is not known a priori. In order to determine the “natural” roll rate of the tail section – the rate at which the spin-averaged rolling moment on the tail is zero – an iterative process is used. First, it’s assumed that the tail rotation rate is low enough that the variation of spin-averaged tail rolling moment with rotation rate is linear. Two rotation rates are then imposed on the tail; the first a reasonable guess and the second intended to provide an opposite sign in averaged tail rolling moment to the first. The resulting spin-averaged tail rolling moments from these two simulations, along with the static results discussed above (i.e. a zero-spin-rate simulation) are then fit with a straight line to determine the natural roll rate of the tail section. A third dynamic simulation is then performed at the natural roll rate in order to confirm the prediction.

An initial guess of 2500 rpm for the tail rotation rate was used. A time-resolution study was performed at this rotation rate using timesteps that move the tail fins

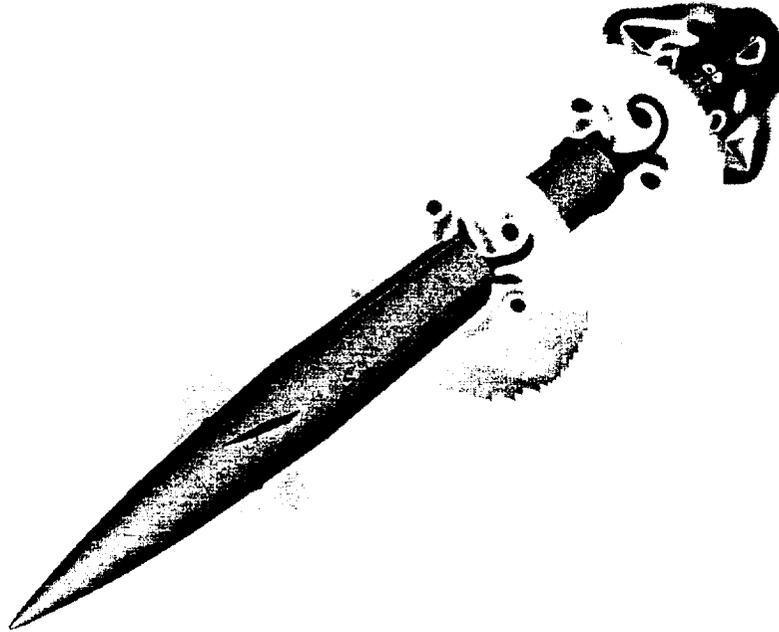


Figure 5: Velocity magnitude contours (blue is low, red is high) for static simulation with tail in \times position ($M_\infty = 1.6$, $\alpha = 4.0^\circ$).

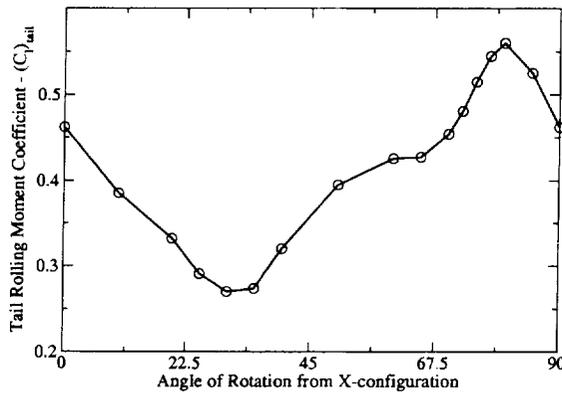


Figure 6: Tail rolling-moment $(C_l)_{tail}$ for static, fixed-tail simulations ($M_\infty = 1.6$, $\alpha = 4.0^\circ$).

2° , 1° , and 0.5° of rotation per step respectively. The results of this time-resolution comparison showed no difference in $(C_l)_{tail}$ vs. rotation angle between the 1° and 0.5° of rotation/timestep simulations. As a result 1° of roll per timestep was utilized for

all of the simulations discussed here.

Figure 7 shows the spin-averaged tail rolling moment against the imposed rotation rate for the iterative process discussed above. The variation of averaged tail rolling moment with rotation rate is confirmed to be linear, and the predicted natural rotation rate is 3165 rpm for these conditions. A final dynamic simulation with the imposed natural rotation rate was performed. The variation of tail rolling moment with rotation angle is shown in Fig. 8 for all computations; the static and the three dynamic with an imposed rotation rate. The simulation with the natural rotation rate does provide zero spin-averaged tail rolling moment. As the vortices, canard downwash, and wind vector do not change when the tail spins, the variation of tail rolling moment with rotation angle is similar for all simulations, however shifted as the rotation rate increases. In other words, the rotation of the tail section provides minor dynamic effects itself, and what effects there are wash downstream without influencing the aerodynamic loads. When the velocity of the tail section "balances" the outer flow effects, a stable spin rate is found.

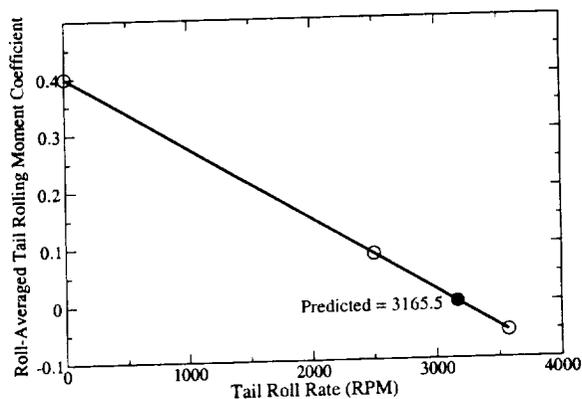


Figure 7: Predicted "natural" tail rotation rate ($M_\infty = 1.6$, $\alpha = 4.0^\circ$).

Velocity magnitude contours through the tail section, viewed from the nose, are shown in Fig. 9 for the natural rotation rate - 3165 rpm. As the fins encounter the vortices there is a strong interaction, however after the fins pass the vortices reform in their original positions. A detailed comparison of the flow features between the dynamic and static simulations has not been performed as yet, however it will be included in the final paper.

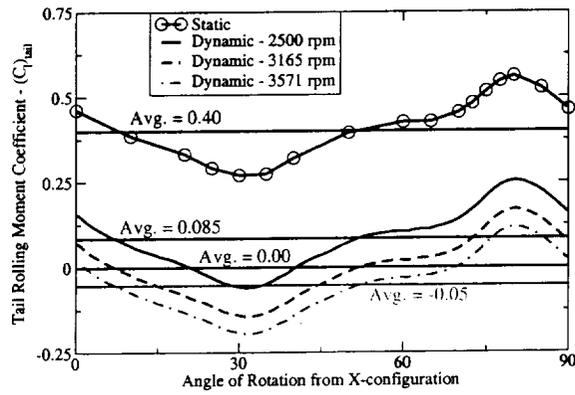
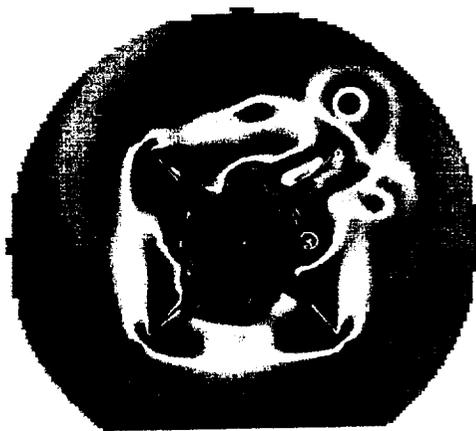
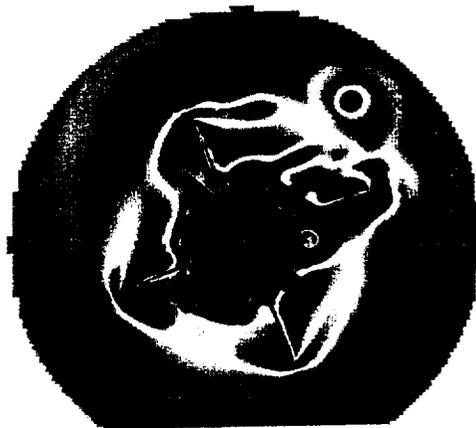


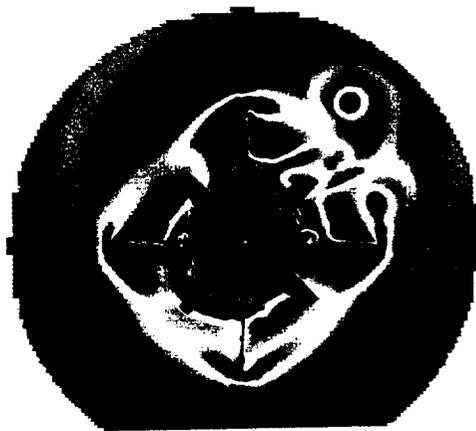
Figure 8: Tail rolling-moment $(C_l)_{tail}$ for static, fixed-tail and dynamic, imposed-rotation simulations ($M_\infty = 1.6$, $\alpha = 4.0^\circ$).



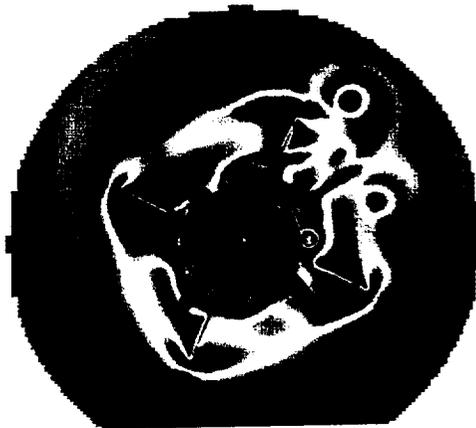
(a) $\phi_{tail} = 0.0^\circ / 90.0^\circ$



(b) $\phi_{tail} = 22.5^\circ$



(c) $\phi_{tail} = 45.0^\circ$



(d) $\phi_{tail} = 67.5^\circ$

Figure 9: Velocity magnitude contours through the tail section (red is high, blue is low) viewed from the nose ($\phi_{tail} = 3165$ rpm, $M_\infty = 1.6$, $\alpha = 4.0^\circ$).

4 Future Work

A 3-D Cartesian method for simulating the general prescribed motion of rigid bodies has been applied to the analysis of a canard-controlled airframe with a spinning tail section. Static simulations with the tail section at various azimuthal angles were performed as a baseline for comparison with the dynamic simulations. The natural zero spin-averaged rotation rate of the tail section was determined by an iterative process, and found to be 3165 rpm.

A major focus for the remaining work is the comparison of the static and dynamic simulations. The variation of body aerodynamic loads with tail position between the static and dynamic simulations is fundamental. Further, the possibility of approximating the dynamic spin-averaged results with a single static simulation is of interest. In order to model the S&C characteristics of the missile spin-averaged data is required, which necessitates CPU-intensive time-dependent, moving-body simulations. This is prohibitive for a production CFD environment, which may require thousands of data points to build an S&C database. It's preferable to approximate the dynamic simulations with an appropriate static configuration, and possibly apply a post-processing correction. The full paper will present an analysis to determine if approximating the dynamic results with a static configuration is feasible, and if so, present a methodology.

References

- [1] Berger, M.J. and LeVeque, R., "Cartesian Meshes and Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations," in *Proceedings of the 3rd International Conference on Hyperbolic Problems*, (Uppsala, Sweden), 1990.
- [2] Quirk, J., "An Alternative to Unstructured Grids for Computing Gas Dynamic Flows Around Arbitrarily Complex Two Dimensional Bodies," ICASE Report 92-7, 1992.
- [3] Karman, S.L. Jr., "Splitflow: A 3D Unstructured Cartesian/Prismatic Grid CFD Code for Complex Geometries," AIAA Paper 95-0343, 1995.
- [4] Melton, J.E., Berger, M.J., Aftosmis, M.J., and Wong, M.D., "3D Applications of a Cartesian Grid Euler Method," AIAA Paper 95-0853, 1995.
- [5] Aftosmis, M.J., Berger, M.J., and Melton, J.E., "Robust and Efficient Cartesian Mesh Generation for Component-Based Geometry," AIAA Paper 97-0196, Jan. 1997. Also *AIAA Journal* **36**(6):952-960, June 1998.
- [6] Wang, Z.J., Przekwas, A., and Hufford, G., "Adaptive Cartesian/Prism Grid Generation for Complex Geometry," AIAA Paper 97-0860, Jan. 1997.

- [7] Bayyuk, S.A., Powell, K.G., and van Leer, B., "Computation of Flows with Moving Boundaries and Fluid-structure Interactions," AIAA Paper 97-1771, 1997.
- [8] Lahur, P.R. and Nakamura, Y., "Simulation of Flow around Moving 3D Body on Unstructured Cartesian Grid," AIAA Paper 2001-2605, June 2001.
- [9] Murman, S.M., Aftosmis, M.J., and Berger, M.J., "Numerical Simulation of Rolling-Airframes Using a Multi-Level Cartesian Method," AIAA Paper 2002-2798, June 2002.
- [10] Murman, S.M., Aftosmis, M.J., and Berger, M.J., "A 3-D Time-Dependent Cartesian Method with Moving Boundaries," AIAA Paper 2003-1119, Jan. 2003.
- [11] Haimes, R. and Aftosmis, M.J., "On Generating High Quality "Water-tight" Triangulations Directly from CAD," in *Meeting of the International Society for Grid Generation (ISGG) 2002*, June 2002.
- [12] Haimes, R. and Follen, G., "Computational Analysis Programming Interface," in *Proceedings of the 6th International Conference on Numerical Grid Generation in Computational Field Simulations*, (University of Greenwich), June 1998.
- [13] Aftosmis, M.J., Delanaye, M., and Haimes, R., "Automatic Generation of CFD-ready Surface Triangulations from CAD Geometry," AIAA Paper 99-0776, Jan. 1999.
- [14] Aftosmis, M.J., Berger, M.J., and Adomavicius, G., "A Parallel Multilevel Method for Adaptively Refined Cartesian Grids with Embedded Boundaries," AIAA Paper 2000-0808, Jan. 2000.
- [15] Merkle, C.L. and Athavale, M., "A Time Accurate Unsteady Incompressible Algorithm Based on Artificial Compressibility," AIAA Paper 87-1137, June 1987.
- [16] Rogers, S.E., Kwak, D., and Kiris, C., "Numerical Solution of the Incompressible Navier-Stokes Equations for Steady-State and Time-Dependent Problems," AIAA Paper 89-0463. Also *AIAA Journal* **29**(4):603-610, 1991.
- [17] Jameson, A., "Time Dependent Calculations Using Multigrid with Applications to Unsteady Flows Past Airfoils," AIAA Paper 91-1596, June 1991.
- [18] Melson, N.D., Sanetrik, M.D., and Atkins, H.L., "Time-accurate Navier-Stokes Calculations with Multigrid Acceleration," in *Proceedings of the Sixth Copper Mountain Conference on Multigrid Methods*, Apr. 1993.
- [19] Zhang, H., Reggio, M., Trépanier, J.Y., and Camarero, R., "Discrete Form of the GCL for Moving Meshes and its Implementation in CFD Schemes," *Computers in Fluids*, vol. 22, no. 1, pp. 9-23, 1993.

- [20] Lesoinne, M. and Farhat, C., "Geometric Conservation Laws for Flow Problems with Moving Boundaries and Deformable Meshes, and Their Impact on Aeroelastic Computations," *Computer Methods in Applied Mechanics and Engineering*, vol. 134, pp. 71–90, 1996.